UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/827,290	04/05/2001	Daniel C. Berg	RSW920000173US1	5722

7590 06/14/2005

Gregory M. Doudnikoff
IBM Corporation T81/503
P.O. Box 12195
Research Triangle Park, NC 27709

EXAMINER	
PHAM, CHRYSTINE	
ART UNIT	**PAPER NUMBER**
2192	

DATE MAILED: 06/14/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/827,290	BERG ET AL.	
	Examiner	Art Unit	
	Chrystine Pham	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *22 February 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,5-7,10,11 and 14-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,5-7,10,11 and 14-18* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on February 22nd 2005 has been entered.

2. Claims 1, 5-7, 10, 11, and 14 have been amended. Claims 2-4, 8, 9, 12, and 13 have been canceled. Claims 15-18 are new claims. Claims 1, 5-7, 10, 11, 14-18 are presented for examination.

Response to Arguments

3. Applicants' arguments filed on February 22nd 2005 in regards to claim rejections have been fully considered but they are not persuasive.

First, the Applicants suggest that Rubin's pointers, which are maintained (i.e., stored) within the source and sink instances, do not teach the claimed "association end reflects association from an instance of a first class to and instance of a second class" and "the association has an inverse association end to reflect an inverse association from the instance of the second class to the instance of the first class" (pages 9-10). The Applicants further seem to suggest that Rubin does not teach "bi-directional links" which characterizes the scenario "where an association has an inverse, and can be navigated in both directions (that is, from the instance of the first class to the instance of the second class and vice versa" (page 10). As has been established in Office Action dated December 22nd 2004, Rubin explicitly teaches binary bi-directional relations (i.e., bi-directional links) (see at least col.3:40-65). In the col.3:40-65 as well as the Abstract, Rubin specifically teaches implementing binary bi-directional relations by creating a source instance from a source class and one or more sink instances from a sink class. The

source instance has a first sink pointer and a last sink pointer, stored within the source instance, pointing to the first sink instance and the last sink instance respectively. The sink instance includes one source pointer, stored within the sink instance, pointing to the source instance. Rubin explicitly discloses using pointers (i.e., sink pointers and source pointers) stored within the source instance and sink instances to create a doubly-linked ring of instances. The doubly-linked ring of instances specifies a relationship between the source instance and one or more sink instances. In other words, the pointers, which are stored within the source instance and one or more sink instances, are used to implement bi-directional relations (i.e., bi-directional links, or associations and inverse associations) between the source instance and one or more sink instances. It is quite clear that the first or last sink pointer, stored in the source instance, enables navigation in direction (of the "association") from the source instance (i.e., instance of a first class) to the sink instance (i.e., instance of a second class). It is quite clear that the source pointer, stored in the sink instance, enables navigation in the inverse direction (of the "inverse association") from the sink instance (i.e., instance of a second class) to the source instance (i.e., instance of a first class). Thus, Rubin clearly and explicitly teaches "bi-directional links which can be navigated in both directions (that is, from the instance of the first class to the instance of a second class, and vice versa)".

Second, the Applicants argue that "in a scenario where there are 3 or more sink instances, Rubin's source instances do not point to all of the sink instances" (page 10), it is submitted this argument is irrelevant to the limitations cited in the claims. Rubin's source instances do not have to point to ALL of the sink instances to anticipate the limitations of the claims as has been established previously. Similarly, the Applicants rely on Rubin's discussion of modifying his invention to include "dummy" sink instances to enable insertion and removal of relationships without requiring access to source instances (i.e., a new sink instance must always be inserted such that it is neither the first sink nor the last sink) to contrast Rubin's teaching and the claim limitations. It is submitted that, Rubin discloses these features (i.e., dummy sink

instances) as way in which his disclosed invention can be modified, thus, these additional features do not limit or teach away from his disclosed invention. As a matter of fact, in col.11:55-60, Rubin specifically states that "source pointer 46 of each sink instance is set to the address of the new source instance". It is clear that Rubin is not always employing the dummy sink instances, which render the dummy sink instances non-essential to his disclosed invention. Furthermore, in his discussion of the dummy sink instances in col.18:42-67, Rubin specifically states that his "invention can be employed in such a way that relationships can be inserted or removed without access to source instances .. **in cases** where a user may have write access to a database containing sink instances, but no access, or only read access to a database containing related source instances". Again, it is clear that Rubin's dummy sink instances are only an addition (thus non-essential) to enable the make and use of his invention **in cases** where a user may have write access to a database containing sink instances, but no access, or only read access to a database containing related source instances.

The Applicants further remark that the Office Action has misinterpreted the "single multiplicity" and "many multiplicity" concept and that Rubin's "presents" relationship has a many multiplicity (because it presents more than one sink instances), not a single multiplicity and that Rubin's "isPresentedBy" relationship is the one that has a single multiplicity (because each sink instance is presented by at most one source instance) (pages 11-12). It is submitted that this characterization of Rubin's teaching contradicts with Applicants' assertion that "Rubin's one-to-many relationships have both single and many multiplicity" (page 12). Thus, rather than the Office Action misinterpreting the concept, Rubin's "presents" relationship has two ends (i.e., source end and sink end) and can be equally attributed or characterized as having a many multiplicity (because it presents more than one sink instance) or having a single multiplicity (because only a single source instance can present one or more sink instances).

Applicants attempt to contrast Applicants' "association" and that of Rubin's by using the concept "upper bound". According to Applicants, "a one-to-many association end" has only the many multiplicity" (page 12). In other words, the many multiplicity in the above one-to-many association end is the "upper bound" that is "considered" by Applicants' invention. Following Applicants' rationale, each association end has only one multiplicity to consider, either that multiplicity is a single multiplicity or a many multiplicity, and not both. It is submitted that concept of "upper bound" is clearly anticipated by Rubin. In the association end (i.e., source instance end) of Rubin's "presents" relation (i.e., one-to-many association), multiple sink pointers are maintained (see at least *SOURCE INSTANCE 26, FIRST SINK, LAST SINK 27* FIG.5 & associated text), that is to say, the one-to-many association end has only the many multiplicity to consider. In the inverse association end (i.e., sink instance), only one source pointer is maintained for each sink instance 28 pointed to by the single source instance 26. Furthermore, each source pointer in each sink instance implements the "isPresentedBy" association between only that sink instance and the source instance. That is to say, the "isPresentedBy" association between each sink instance and the source instance is a "one-to-one" association (one sink isPresentedBy one source). Thus, the sink instance end is at once, an inverse association end of the one-to-many "presents" association, AND the "one-to-one" association end of the association "isPresentedBy" which has only the single multiplicity (i.e., single source pointer) to consider.

In view of the fore going discussion, previous rejection of claims under 35 U.S.C. 102(b) and 103(a) is considered proper and maintained.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1, 6, 7, and 11 are rejected under 35 U.S.C. 102(b) as being anticipated by Rubin of record, hereinafter, *Rubin.*

Claim 1

Rubin teaches a system (e.g., see fig.1 & associated text) and method for programmatically enforcing referential integrity constraints (e.g., col.5:30-47; see *referential integrity* col.6:35-37) among associations (e.g., see *presents 20* fig.2 & associated text) between class instances, comprising the steps of:

- determining, when evaluating (e.g., col.11:59-64) a request to set an association end to reflect an association (e.g., see *presents 20* fig.2 & associated text; see *presents* fig.3 & associated text; see *ISPRESENTEDBY 40, TOONE 41, SINK 24* fig.4 & associated text) from an instance of a first class (e.g., see *SOURCE INSTANCE 26* fig.3 & associated text; see *SOURCE 22* fig.2 & associated text; see *SINK INSTANCE 28* fig.3 & associated text; see *SINK 24* fig.2 & associated text) to an instance of a second class (e.g., see *SOURCE INSTANCE 26* fig.3 & associated text; see *SOURCE 22* fig.2 & associated text; see *SINK INSTANCE 28* fig.3 & associated text; see *SINK 24* fig.2 & associated text) whether the association end has a single multiplicity (e.g., see *ISPRESENTEDBY 40, TOONE 41, SINK 24* fig.4 & associated text; see *source pointer 46* fig.5 & associated text) **or** a many multiplicity (e.g., see *presents* fig.3 & associated text; see *FANPRESENTS 31, FAN 30, SOURCE 22* fig.4 & associated text; *FIRST SINK, LAST SINK 27* fig.5 & associated text);
- if the association end to be set has the single multiplicity (e.g., see *ISPRESENTEDBY 40, TOONE 41, SINK 24* fig.4 & associated text; see *source pointer 46* fig.5 & associated text), atomically and programmatically performing the steps of:
- setting (i.e., programmatically modifying) an inverse association end of the association to reflect an inverse association (i.e., source *presents* sink) (e.g., see *presents 20* fig.2 & associated text)

from the instance of the second class (i.e., source instance) to the instance of the first class (i.e., sink instance) (e.g., col.2:40-43; see *inserting sink instance into a doubly-linked ring of sink instances* col.2:52-56; see *74, 76* fig.7 & associated text), after disconnecting the inverse association end from an existing instance of the second class, if any (e.g., see *relationships, source pointer* col.3:4-9; see *setting the source pointer to the address of the source instance* col.2:52-56; *source pointer 46, null* col.9:40-45; see *78* fig.7 & associated text);

- setting the requested association end from the instance of the first class to the instance of the second class (e.g., col.2:40-43 & col.2:52-56; see *80* fig.7 & associated text); and
- if the association end to be set has the many multiplicity (e.g., see *presents* fig.3 & associated text; see *FANPRESENTS 31, FAN 30, SOURCE 22* fig.4 & associated text; *FIRST SINK, LAST SINK 27* fig.5 & associated text), atomically and programmatically performing the steps of:
    - adding the requested association end to the instance of the first class (i.e., source instance) (e.g., see *74, 76* fig.7 & associated text); and
    - setting (i.e., programmatically modifying) an inverse association end of the association to reflect an inverse association (i.e., sink isPresentedBy source) (e.g., see *ISPRESENTEDBY 40, TOONE 41, SINK 24* fig.4 & associated text; see *source pointer 46* fig.5 & associated text) from the instance of the second class (i.e., sink instance) to the instance of the first class (i.e., source instance) (e.g., see *80* fig.7 & associated text), after disconnecting the inverse association end from an existing instance of the first class, if any (e.g., *source pointer 46, null* col.9:40-45; see *78* fig.7 & associated text);

Claim 6

Rubin teaches a method according to claim 1, wherein the method is provided as link helper a single link helper object and a multiple link helper object for each association, wherein the single link helper object performs the atomically and programmatically performed steps for the single multiplicity association end (e.g., see fig.5 *39* & associated text) and the multiple link helper object performs the

atomically and programmatically performed steps for the many multiplicity association end (e.g., see fig.5 *27* & associated text; col.7 : 10-32).

Claim 7

Rubin teaches a computer program product for programmatically (e.g., col.5 : 31-35 & 40-45) enforcing referential integrity constraints (e.g., col.6 : 35-37) among associations (e.g., see fig.2 *presents relation 20* & associated text) between class instances (e.g., col.6 : 25-31), wherein the computer program product is embodied on one or more computer readable media (e.g., fig.1 *storage device 16* & associated text; see *disk* col.6 : 19-23) and comprises computer-readable code means for performing the steps as addressed in claim 1 (see claim 1).

Claim 11

Rubin teaches a system (e.g., fig.1 & associated text) for performing the method as addressed in claim 1 (see claim 1).

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 5, 10, and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Rubin* in view of Johnson (http://www.javaworld.com/javaworld/jw-02-1998/jw-02-beans.html) (hereinafter *Johnson*).

Claim 5

Rubin teaches a method as applied to claim 1, further comprising step of

- determining whether the association end to be set, or the inverse association end is a primary end of the association (e.g., see fig.4 *ispresentedby 40* & *toone 41*; fig.5 *46;* associated text; col.10 : 66 - col.11 : 6 & 59-64) but fails to teach serializing only the primary end of the association during a serialization operation.

However, *Johnson* teaches a method of selectively serializing JAVA objects (classes, instances of classes, fields), that is to say, serializing of only the primary end of the association during a serialization operation (e.g., see section *Serial killers: How to avoid unwanted serialization,* pg. 12, 2nd par.-4th par.). Therefore, one of ordinary skill in the pertinent art, at the time the invention was made, would have been motivated to modify the teaching of *Rubin* to include serialization of the association's primary end as to transform it into persistent data which can be stored, transmitted, recreated, and retrieved at another place and time.

Claims 10, 14

Claim recites limitations, which have been previously addressed in claim 5, therefore is rejected for the same reasons as cited in claim 5.

8. Claims 15-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Rubin* in view of Lindberg et al. (*Lindberg*, US 2003/0028540 A1).

Claim 15

The rejection of base claim 1 is incorporated. *Rubin* does not expressly disclose wherein in one or more structured markup language representations specify instances of the first class, instances of the second class, and associations between the instances of the first and second classes. However, *Lindberg* discloses a method of enforcing referential integrity constraints (see at least paragraph [0085]) one or more structured markup language representations specify instances of the first class, instances of the second class, and associations between the instances of the first and second classes (see at least

information model entities, XML documents, item elements, relationship elements, entities paragraph [0012]; *business object instances, XML document* paragraph [0016]; paragraphs [0106]-[0113]). *Rubin* and *Lindberg* are analogous art because they are both directed to enforcing referential integrity constraints among associations between class instances. It would have been obvious to one of ordinary skill in the pertinent art at the time the invention was made to incorporate the teaching of *Lindberg* into that of *Rubin* for the inclusion of a structured markup language representations specifying class instances and associations between them. And the motivation for doing so would have been to enforce referential integrity constraints in creating, updating, or removal of new class instances without requiring extensive coding (see *Lindberg* paragraphs [0081]-[0085]).

Claim 16

The rejection of base claim 15 is incorporated. *Lindberg* further teaches wherein only one association end for each association between instances is specified in the structured markup language representations (see at least *TABLE 1, TABLE 3, Entity Definition, Country, Role, Relationship, countryArea, timezones, remoteMultiplicity* paragraphs [0042]-[0045]).

Claim 17

The rejection of base claim 16 is incorporated. *Lindberg* further teaches wherein the only one association end is an association end designated as a primary end for the association (see at least *Relationship element, entities, navigationName, remoteMultiplicity* paragraphs [0069]-[0073]).

Claim 18

The rejection of base claim 15 is incorporated. *Lindberg* further teaches wherein a serialization of results of the request to set the association end that has the single multiplicity (see at least) comprises the step of:

- determining whether to association end to be modified/set is a primary end for the association (see at least *TABLE 3, Entity Definition, Country, Role, Relationship,*

countryArea, timezones, remoteMultiplicity paragraph [0045]; *Relationship element, entities, navigationName, remoteMultiplicity* paragraphs [0069]-[0073]), and if so, programmatically performing the steps of:

- removing the presentation of the previously-existing inverse association end, if any, from the structured markup language representation in which it is specified (see at least *actions, object, delete* paragraphs [0081]-[0083]; *readOnly, false* paragraphs [0103]-[0104]; *referential integrity constraints, entities, relationships* paragraphs [0084]-0085]); and
- adding a structured markup language representation of the new inverse association end (see at least *referential integrity constraints, entities, relationships* paragraphs [0084]-0085]).

Conclusion

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chrystine Pham whose telephone number is 571-212-3702. The examiner can normally be reached on Mon-Fri, 8:30am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

CP
June 1, 2005

TUAN DAM
SUPERVISORY PATENT EXAMINER